REMARKS

This Response is submitted in response to the Final Office Action mailed September 24, 2008 and the Advisory Action mailed January 9, 2009.

Claims 1, 10, 28, 32, 35, 39, and 41 are amended. New claims 42-46 are presented. Claims 1-18, 28, 32, 34-36, and 39-46 are pending in the Application. Reconsideration of the pending claims is respectfully requested in view of the following remarks.

I. Summary of Office Action

The objection to claims 32 and 34 has been withdrawn in view of Applicants' amendment filed August 27, 2008.

Claims 1-18 and 39 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over U.S. Patent No. 6,922,708 issued to Sedlar ("Sedlar"), in view of U.S. Patent No. 7,275,063 issued to Horn ("Horn").

Claims 28 and 34-36 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over U.S. Patent No. 6,625,624 issued to Chen ("Chen"), in view of U.S. Patent No. 6,356,902 issued to Tan ("Tan").

Claim 32 stands rejected under 35 U.S.C. § 103(a) as being unpatentable over Chen, in view of Tan, and further in view of Sedlar.

Claim 40 stands rejected under 35 U.S.C. § 103(a) as being unpatentable over Sedlar, in view of Horn, and further in view of U.S. Patent No. 6,636,250 issued to Gasser et al. ("Gasser").

Claim 41 stands rejected under 35 U.S.C. § 102(e) as being anticipated by Horn.

II. Claims Rejected Under 35 U.S.C. § 102

With respect to the § 102(e) rejection of independent Claim 41 over Horn, without conceding the merits of this rejection, Applicants have amended Claim 41 to clarify the

claimed subject matter and advance prosecution. Applicants submit that Horn does not identically disclose any of the following features of Claim 41, as amended: "assigning unique identification numbers to directories of the storage server in a depth first search order during the file walk," "wherein the unique identification numbers correspond to the depth first search order in which the directories are accessed during the file walk," or "traversing the data structure based on the unique identification numbers to determine relationships between the directories of the storage server."

Horn discloses a system for automatically organizing, indexing, and viewing information from multiple sources. For each information object, Horn creates and stores an internal representation of the object in a database, along with an internal unique identifier for the object (col. 8, lines 1-12). Horn determines paths between the stored objects and displays hierarchies depicting these paths (col. 32, line 18 - col. 33, line 33).

However, Horn does not identically disclose assigning unique identification numbers in a depth first search order. Horn describes a thread that recursively descends a hierarchy and creates an array of data entries (col. 34, lines 37-40). Horn does not disclose how its hierarchy is recursively traversed, and in particular does not disclose that its hierarchy is recursively traversed "in a depth first search order." Simply recursively descending a hierarchy does not identically disclose using a depth first search order, which traverses down to the end of a hierarchy first, and then across the hierarchy. Nor is Horn's inclusion of the depth of a folder in a data entry the same as or equivalent to using a depth first search order.

Nor does Horn identically disclose unique identification numbers that correspond to the order in which the directories are accessed during the file walk. Horn describes that each of its data entries includes a file system specifier that represents a file, a depth in the folder hierarchy, and a flag that determines whether the file system specifier is for a folder or a file (col. 34, lines 37-42). However, Horn does not disclose how its file system specifier represents a file, and in particular does not disclose a file system specifier that corresponds to an order in which the file was accessed during a

file walk. Nor is Horn's inclusion of the depth of a folder in a data entry the same as or equivalent to a unique identification number that corresponds to the order in which the file was accessed during a file walk.

In addition, Horn fails to identically disclose traversing a data structure based on the unique identification numbers. Once Horn's data entries are created, as described above, the entries are annotated with metadata, such as file and folder comments. The metadata is then used to classify the data entries, or determine to which collections the data entries belong. (Horn, col. 34, lines 45-58.) However, Horn merely describes classifying data entries based on metadata; Horn does not disclose classifying data entries based on its file system specifier or any unique identification number.

III. Claims Rejected Under 35 U.S.C. § 103

With respect to the § 103(a) rejections of independent Claims 1, 10, and 39 over Sedlar and Horn, without conceding the merits of these rejections, Applicants have amended Claims 1, 10, and 39 to clarify the claimed subject matter and advance prosecution. Applicants submit that Sedlar and Horn do not teach or suggest, either individually or in combination, any of the following features of Claims 1, 10, or 39, as amended: "using a first thread to ... assign a first unique identification (ID) number to a first determined directory and a second unique ID number to a second determined directory, wherein the ID numbers are assigned while the directory structure is being traversed in the DFS order," "wherein the ID numbers correspond to [or depend on, or are chronologically assigned in numerical order based on] the DFS order in which the determined directories are traversed," or "using a second thread to examine the determined files."

The Examiner acknowledges that Sedlar does not teach these recited features. (Final Office Action, Sept. 24, 2008, pp. 7-8.) The Examiner relies upon Horn to cure this deficiency. As described above, Horn discloses a system for automatically organizing, indexing, and viewing information from multiple sources. For each information object, Horn creates and stores an internal representation of the object in a

database, along with an internal unique identifier for the object (col. 8, lines 1-12). Horn determines paths between the stored objects and displays hierarchies depicting these paths (col. 32, line 18 - col. 33, line 33).

However, Horn does not disclose or suggest assigning its unique identifiers while a directory structure is being traversed in a depth first search order. As discussed above, while Horn describes a thread that recursively descends a hierarchy, Horn does not disclose how its hierarchy is recursively traversed, and in particular does not disclose that its hierarchy is recursively traversed "in a depth first search order." Simply recursively descending a hierarchy does not identically disclose using a depth first search order, which traverses down to the end of a hierarchy first, and then across the hierarchy. Nor is Horn's inclusion of the depth of a folder in a data entry the same as or equivalent to using a depth first search order.

Nor does Horn identically disclose unique identification numbers that correspond to the order in which the directories are traversed. Horn describes that each of its data entries includes a file system specifier that represents a file, a depth in the folder hierarchy, and a flag that determines whether the file system specifier is for a folder or a file (col. 34, lines 37-42). However, Horn does not disclose how its file system specifier represents a file, and in particular does not disclose a file system specifier that corresponds to an order in which the file was traversed. Nor is Horn's inclusion of the depth of a folder in a data entry the same as or equivalent to a unique identification number that corresponds to the order in which the file was traversed.

In addition, Horn does not disclose or suggest using a first thread to assign unique identification numbers and a second thread to examine determined files. In fact, Horn describes that a single thread – its scanner thread – performs the function the Examiner believes corresponds to assigning unique identifiers ("Traverse") and the functions the Examiner believes correspond to examining the determined files ("Annotate," "Create," "Classify," and "Notify") (col. 34, lines 34-65).

Independent Claims 1, 10, and 39 are allowable for at least the foregoing reasons.

With respect to the § 103(a) rejection of independent Claim 28 over Chen and Tan, without conceding the merits of this rejection, Applicants have amended Claims 28 to clarify the claimed subject matter and advance prosecution. Applicants submit that Chen and Tan do not teach or suggest, either individually or in combination, any of the following features of Claim 28, as amended: "assigning a depth first search (DFS) ID to the first directory, wherein the DFS ID is assigned while the directory structure is being traversed in a DFS order," "wherein the DFS ID is a unique ID that corresponds to the DFS order in which the first directory is traversed," or "placing the first subset of files in a file queue for examination by a file thread."

Chen discloses a system for archiving and retrieving web pages. Chen describes that different threads, including a user thread, an agent thread, and a walking thread, may be used within its system (Fig. 2). Chen's agent thread requests a web page from a remote server and may provide the web page to its walking thread (col. 3, lines 54-60; Fig. 2). Chen's walking thread may walk through a document hierarchy and perform various functions on a web page, including archiving the web page, searching for keywords, and creating index tables (col. 3, line 59 - col. 6, line 4).

Chen does not disclose assigning a depth first search (DFS) ID to a directory while the directory structure is being traversed in a DFS order. The Examiner indicates that Chen implicitly teaches a depth first search (Advisory Action, Jan. 9, 2009, p. 2). A prior art reference only inherently discloses a claim feature if the system described in the reference *necessarily* functions in accordance with the claim feature (*In re King*, 801 F.2d 1324, 1326 (Fed. Cir. 1986)). However, Chen does not necessarily function in accordance with a depth first search. Chen does not describe how its walking function is performed, and in particular does not describe that its walking function is performed according to a depth first search. Chen's walking function could be performed according to a breadth first search or any other of a number of search techniques.

Accordingly, Chen does not inherently teach a depth first search. Even if Chen were to teach a depth first search, Chen fails to disclose assigning a depth first search (DFS) ID to a directory while the directory structure is being traversed in a DFS order.

Furthermore, Chen fails to disclose "wherein the DFS ID is a unique ID that corresponds to the DFS order in which the first directory is traversed." As described above, Chen fails to disclose a depth first search order. Accordingly, Chen cannot disclose unique identifiers that correspond to such a DFS order.

Nor does Chen disclose or suggest placing a subset of files in a queue for examination by a file thread. While Chen describes using different threads, none of these threads corresponds to "a file thread" that examines a subset of files in a file queue. While the Examiner believes that Chen's agent thread corresponds to Applicants' file thread, the functions the Examiner cites in support are performed by Chen's walking thread, not Chen's agent thread. At most Chen's walking thread may walk through a document hierarchy and perform various functions on a cache of a web page, i.e., a copy of the web page stored in readily accessible memory. The functions performed on the cache of the web page may include archiving the web page, searching for keywords, and creating index tables. (Chen, col. 5, line 59 - col. 6, line 4). However, Chen does not disclose a queue at all, and in particular does not disclose "placing [a] subset of files in a file queue for examination" by a thread.

Nor does Tan disclose any of the following features of Claim 28, as amended: "assigning a depth first search (DFS) ID to the first directory, wherein the DFS ID is assigned while the directory structure is being traversed in a DFS order," "wherein the DFS ID is a unique ID that corresponds to the DFS order in which the first directory is traversed," or "placing the first subset of files in a file queue for examination by a file thread."

Tan discloses mapping a tree structure to a graph map composed of a single-link node. As part of this mapping process, tree nodes at a current level are stored in

"Parent-stack memory" and nodes at the next level are stored in "Child-stack memory" (col. 5, lines 31-33).

However, Tan does not disclose assigning a depth first search (DFS) identifier while a directory structure is being traversed in a DFS order. At most, Tan assigns level numbers to nodes in the tree structure; several nodes may share the same level number (col. 5, lines 25-39; Fig. 2A; Fig. 7). Nowhere does Tan describe that these level numbers are assigned in a depth first search order. Only after the level numbers have been assigned and the tree structure has been mapped to a graph map composed of a single-link node, does Tan perform depth-first or breadth-first searches on the tree structure (col. 7, lines 22-59).

Nor does Tan disclose a unique identifier that corresponds to the DFS order in which a directory is traversed. As described above, Tan fails to disclose that its level numbers are assigned in a depth first search order. Accordingly, Tan cannot disclose unique identifiers that correspond to such a DFS order. Even if Tan were to disclose assigning level numbers in a DFS order, each of Tan's level numbers corresponds to several nodes; Tan's level numbers are not unique identifiers.

Nor does Tan disclose "placing the first subset of files in a file queue for examination by a file thread." Indeed, the Examiner does not cite Tan as disclosing this feature.

Independent Claim 28 is allowable for at least the foregoing reasons.

IV. Conclusion

Thus, for at least the foregoing reasons, the independent claims and all claims that depend from them are believed to be patentable over the applied art.

Please charge any deficiencies or credit any overpayments to our Deposit Account No. 50-2207, under Order No. 672728062US1 from which the undersigned is authorized to draw.

Dated: 2/24/09

Respectfully submitted,

By _____

Judy M. Kadoura
Registration No.: 59,883
P.O. Box 1247
Seattle, Washington 98111-1247
(206) 359-8000
(206) 359-9000 (Fax)
Attorney for Applicant